Data Analysis using a Weak Constraint: The Lorenz Equations

The Lorenz equations give a beautiful illustration of the way in which complex structures emerge from non-linear dynamical systems. All trajectories converge onto the attractor, but within the attractor all trajectories are diverging from each other. The attractor is illustrated in the background. The red colour indicates the region of fastest divergence of nearby trajectories, and dark blue indicates regions where trajectories are converging.

This system can be used to test the ability of the assimilation algorithm to deal with strong non-linearity and to reproduce non-linear structures inherent in the dynamical system.

$$
\begin{aligned}
x_t - a(y - x) &= f, \\
y_t - bx + y + xz &= g, \\
z_t + cz - xy &= h,
\end{aligned}
$$

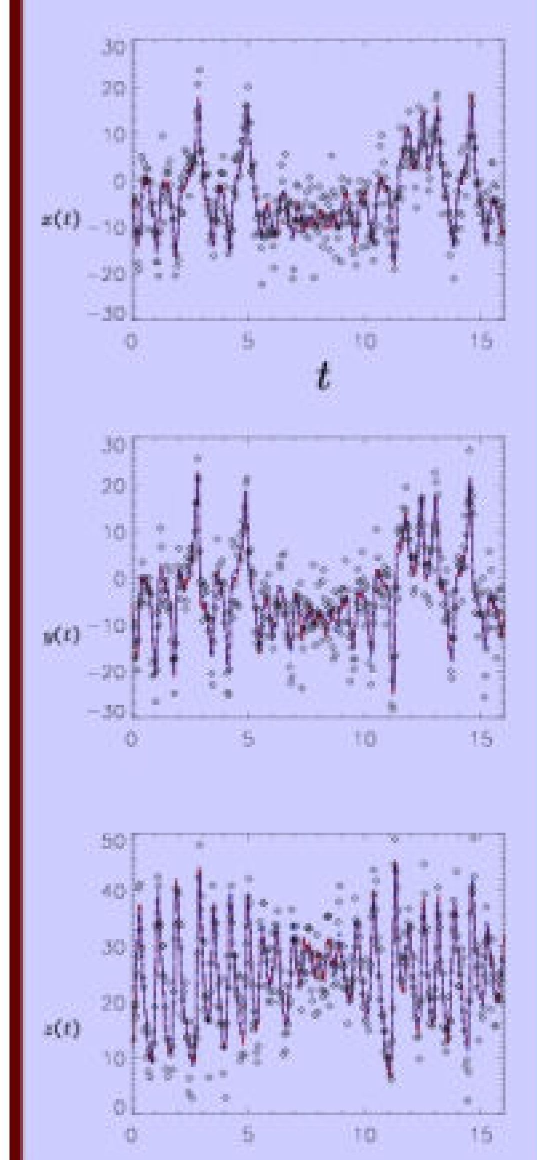
where $a = 10$, $b = 28$, $c = 8/3$ are the constants used by Lorenz (1963) and f, g and h are forcing terms.

The Lorenz system is used here to investigate an approach to data analysis that employs the dynamical equations as a weak constraint. Four key aspects of data analysis are illustrated: **(1) smoothing** noisy observations, **(2) interpolating** between sparse observations, **(3) inferring** unobserved variables and **(4)** deriving **confidence** estimates.

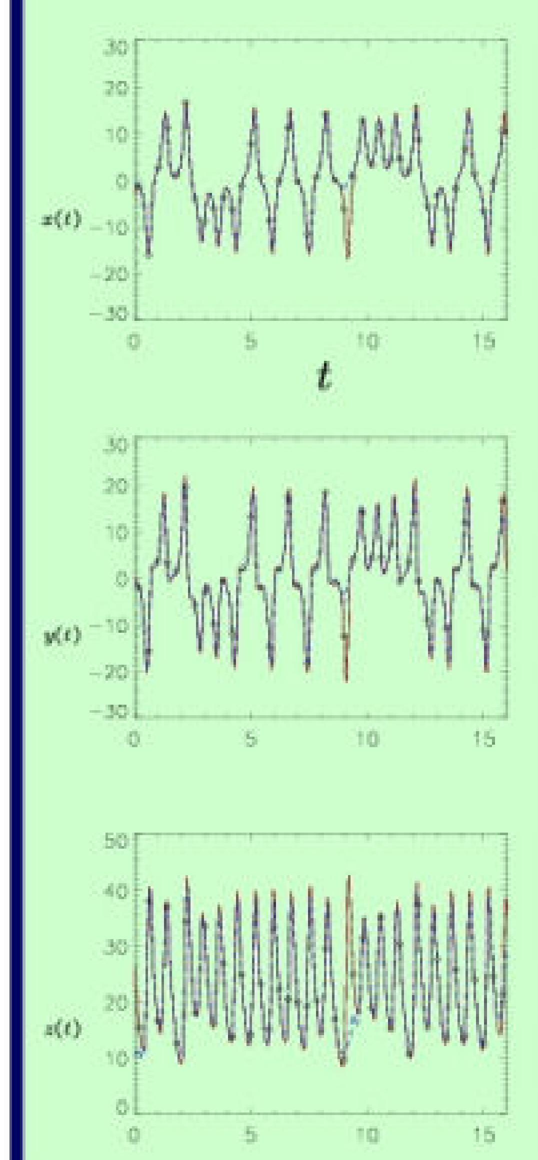
A "true" trajectory is created by integrating the equations with a random, uncorrelated, normally distributed, white noise forcing. The amplitude is defined through the variance the white noise process has when integrated over one time unit, σ_f^2. Observations are then taken, again with uncorrelated, normally distributed noise, with variance σ_o^2. The examples with many observations use $\sigma_f^2 = \sigma_o^2 = 21.3$, while those with sparse observations use $\sigma_f^2 = \sigma_o^2 = 1/12$.

The algorithm used is described in the right hand panels, the results are displayed below.

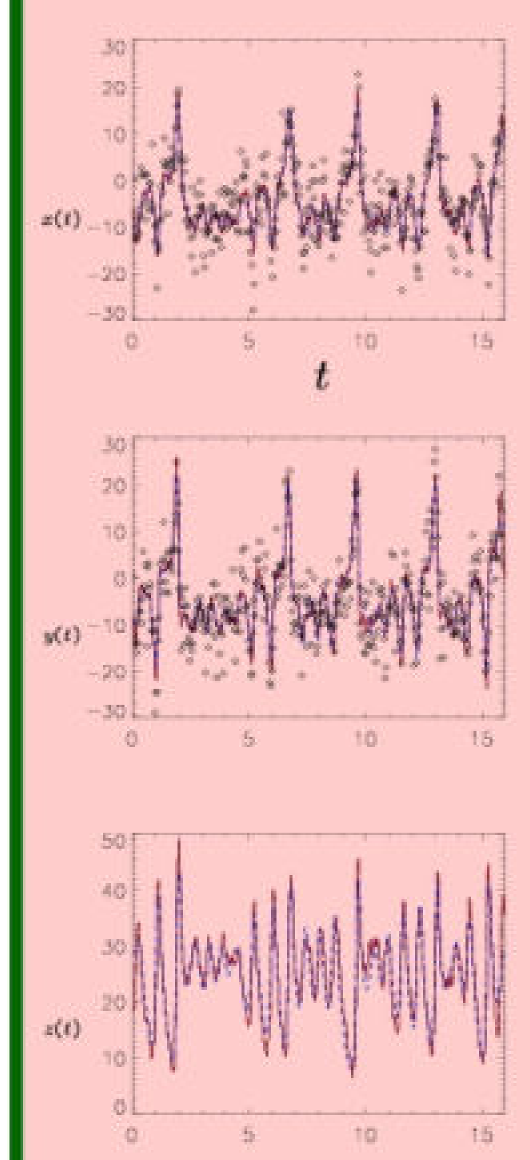
In the first set of experiments x, y and z are "observed" at regular intervals. The figure below shows the three components of the solution, with the true solution in red, the observations markd as squares and the analysis in blue. The evolution of x, y and z is well reproduced in the solution found by minimising the cost function.



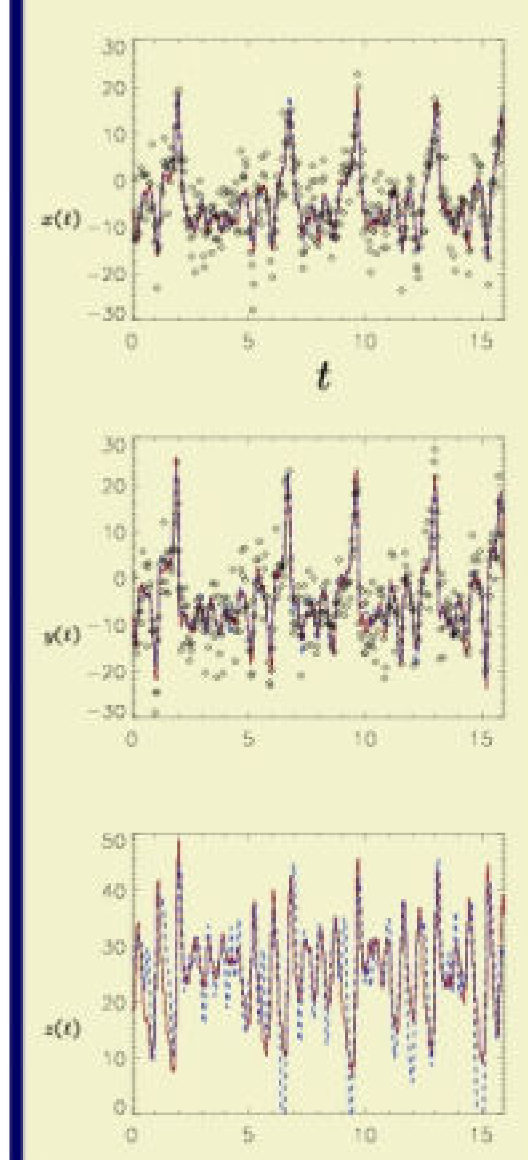
The second set of experiments degrades looks at sparse observations with low observational noise. The analysis is successful, but these results are less robust.
In the previous set of experiments the a priori and observational components of the penalty function were weighted according to the inverse of the error variance. In the present case, using this theoretically optimal choice of weighting does not produce the best results. With such sparsely distributed observations there are multiple minima to the penalty function, and the algorithm has a strong tendency to hang in one of them. If there is no dynamical component the penalty function would be quadratic and hence have a unique minimum. Here a compromise has been found, in which the a priori component is down-weighted by a factor 10. This reduces the tendency to gravitate towards artificial local minima, but still preserves enough information to produce a useful analysis.



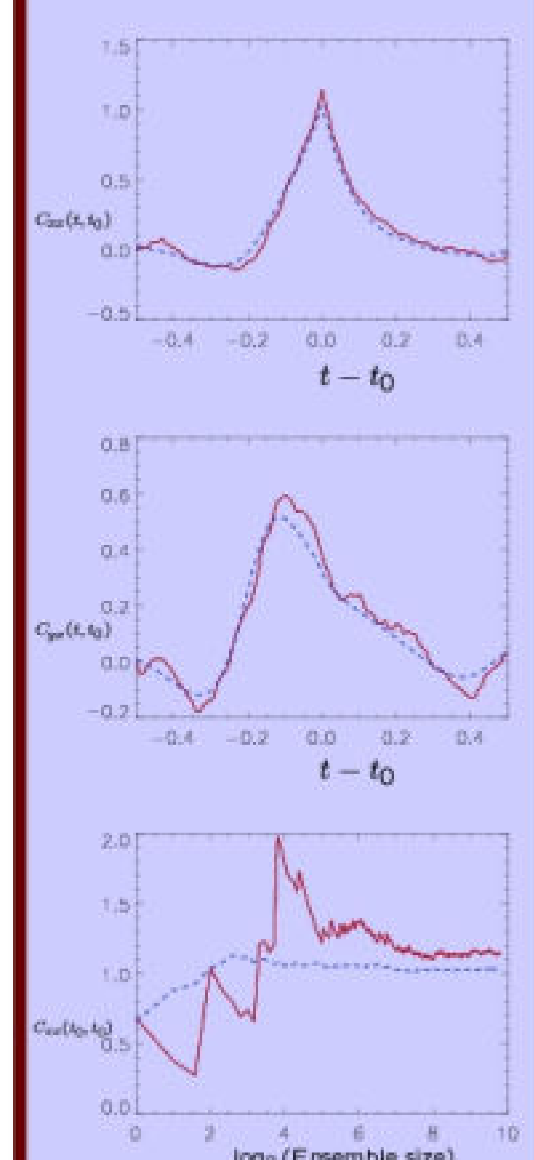
In this example, both x and y observations are used, and the full cost function. Ths analysis is able to infer the evolution of the z component with good accuracy.



In this case there are no z observations, and the reduced penalty function $\mathcal{J}^*_{\rm ap}$ is used, in which the z-evolution equation is omitted. In this case, the only source of information about the z component is the xz term in the y-evolution equation. It can be seen, however, that the analysis algorithm is still able to extract some information about the z trajectory. This shows that the y-equation should not be thought of simply as a constraint on the y-component of the analysis. Here it is providing information about the z component.



In order to illustrate the estimation of the error covariance, an ensemble of 160 analyses has been used. There is some flow dependence in the covariance structure, so the results shown are for a composite of points at which the trajectory passes from negative to positive x. There are around 900 such zero crossings in the 160 analyses.
The upper two graphs show the lagged error covariance between the x error and itself ($C_{xx}(t,t_0) = E(x_e(t)x_e(t_0))$, where t_0 is taken as the time of the zero crossing), and the covariance between x and y errors ($C_{yx}(t,t_0)$). The red line is evaluated from the actual errors and the blue dashed line shows the estimate from the sensitivity of the analysis. The small discrepancy appears to be associated with the numerical errors in the time discretisation. The time asymmetry in C_{yx} is a clear indication of the flow dependence in the error covariance structure.
The lower graph shows how $C_{xx}(t_0,t_0)$ converges onto its ultimate value as the ensemble size is increased. A large ensemble is needed for the direct calculation, but the sensitivity based estimate converges after around 10 members are included.
show the



The Penalty Function

The analysis is defined as the minimum of a penalty function. This is equivalent to the maximum of the joint probability distribution function under the assumption of independent, normally distributed, random errors in the observations and model equations. The penalty function can be split up as follows:

$$\mathcal{J} = \mathcal{J}_{\rm ap} + \mathcal{J}_{\rm obs} + \mathcal{J}_{\rm num}$$

where $\mathcal{J}_{\rm ap}$ represents *a priori* knowledge, $\mathcal{J}_{\rm obs}$ is an observational component, and $\mathcal{J}_{\rm num}$ is there to guarantee good behaviour in finite difference approximations. $\mathcal{J}_{\rm num}$ does not play a significant role here, but becomes important when the method is applied to partial differential equations.

The *a priori* penalty

The a priori penalty function used for the "full dynamics" analyses is:

$$\mathcal{J}_{\rm ap} = w_{\rm ap}\left[(x_t - a(y-x))^2 + (y_t - bx + y + xz)^2 + (z_t + cz - xy)^2\right].$$

The effect of having an incomplete set of evolution equations is investigated by using the reduced penalty function

$$\mathcal{J}^*_{\rm ap} = w_{\rm ap}\left[(x_t - a(y-x))^2 + (y_t - bx + y + xz)^2\right].$$

Note that, although the z-evolution equation has been omitted, there is still some z-dependence in the penalty function, through the term in the y evolution equation.

The analysis algorithm

The penalty function is minimised by an iterative method. At each iteration the value of x, y and z are adjusted at each point according to a local minimisation criterion. That is, each point is adjusted so that the penalty function would be minimised if all other points were held fixed. This approach is known as the "relaxation" algorithm in the context of numerical solutions of elliptical partial differential equations and as "Iterated Conditional Modes" in image analysis. The implementation used for the Lorenz equations is not very efficient in terms of computational resources required, but some work (to be reported) with partial differential systems shows that using a multigrid method reduces the overall cost to $O(N \ln N)$, where N is the total number of degrees of freedom in the system (number of variables $\times$ number of spatial grid points $\times$ number of time steps). This makes it feasible to apply the method to large systems.

The analysis error covariance

The value of an analysis is greatly enhanced if it is possible to quantify the associated error magnitudes. If the penalty function is based on accurate models of the independent errors in the dynamical equations and in the observations, then the formulation is such that the probability distribution of the system trajectory is given by

$$\mathcal{P} \propto \exp\left[-\mathcal{J}\right].$$

If we assume that the analysis errors are sufficiently small that we can linearise the penalty function about the analysis, then it follows that

$$\mathcal{P} \propto \exp\left[\frac{-\mathbf{x}_e^t C^{-1} \mathbf{x}_e}{2}\right],$$

where $\mathbf{x}_e$ the error in the state vector, made up of all variables at all time points, C is the error covariance and the t superscript indicates a transpose.
For small amplitude changes in observations, the resulting change in the analysis is given by

$$\delta \mathbf{x}_{\rm a} = C \delta \mathbf{x}_{\rm obs},$$

where $\mathbf{x}_{\rm obs}$ is an observational state vector obtained by embedding the observations in the model state vector. Direct evaluation of such a matrix equation would be intractable for large systems, but the present methodology allows the equation to be solved indirectly (as alluded to above), exploiting the sparse structure in C^{-1}. This means that the sensitivity of the analysis can be used to estimate the analysis error covariance.

Martin Juckes, BADC, RAL: UGAMP, April 2002